**DML MINIPROJECT**

**CODE:**

import tensorflow.keras as keras

import tensorflow as tf

print(tf.\_\_version\_\_)

import tensorflow.keras as keras

import tensorflow as tf

print(tf.\_\_version\_\_)

print(x\_train[0])

import matplotlib.pyplot as plt

plt.imshow(x\_train[0],cmap=plt.cm.binary)

plt.show()

x\_train = tf.keras.utils.normalize(x\_train, axis=1)

x\_test = tf.keras.utils.normalize(x\_test, axis=1)

model = tf.keras.models.Sequential()

model.add(tf.keras.layers.Flatten())

model.add(tf.keras.layers.Dense(128, activation=tf.nn.relu))

model.add(tf.keras.layers.Dense(128, activation=tf.nn.relu))

model.add(tf.keras.layers.Dense(10, activation=tf.nn.softmax))

model.compile(optimizer='adam',

              loss='sparse\_categorical\_crossentropy',

              metrics=['accuracy'])

model.fit(x\_train, y\_train, epochs=3)

val\_loss, val\_acc = model.evaluate(x\_test, y\_test)

print(val\_loss)

print(val\_acc)

predictions = model.predict(x\_test)

print(predictions)

predictions = model.predict(x\_test)

print(predictions)

plt.imshow(x\_test[0],cmap=plt.cm.binary)

plt.show()

**OUTPUT:**

